Cyclic Integer Sequences as Time-Varying Approximations of π

A Novel Method for Float-Free π Substitution in Array Computation

Franz Kruhm · 2026

Abstract

This paper presents a novel method for approximating the mathematical constant π using a finite cyclic sequence of integers, applied per-element across an array. Rather than treating π as a fixed constant, the method assigns a different integer approximation to each position in the array by cycling through a predetermined digit sequence. The proposed sequence [2, 2, 2, 2, 8, 4, 2] is demonstrated to yield a cycle mean of 22/7 — a classical rational approximation of π accurate to six significant figures — while operating entirely within integer arithmetic. This approach is of particular interest in constrained computational environments where floating-point arithmetic is unavailable or undesirable.

1. Introduction

The constant π appears throughout mathematics, signal processing, and scientific computing. In most contexts it is treated as an immutable scalar value, approximated to sufficient precision using floating-point representation. However, on embedded systems, microcontrollers, and digital signal processors lacking hardware floating-point units, this assumption does not hold. Integer-only environments require alternative strategies for computations that would ordinarily depend on π.

Existing approaches include fixed-point arithmetic — representing π as a scaled integer such as 355 with an implicit denominator of 113 — and lookup table methods that precompute trigonometric values offline. These methods are well-established but treat π as a static quantity. The method proposed here takes a different approach: rather than fixing π at a single value, it allows the effective π approximation to vary across array positions by cycling through a sequence of integers whose aggregate statistical behavior converges on π.

This yields a time-varying π — one that is locally imprecise but globally coherent — opening a novel design space for integer-only computation and generative signal methods.

2. Method

2.1 The Cyclic Digit Sequence

The proposed approximation is defined by the sequence D = [2, 2, 2, 2, 8, 4, 2] of length L = 7. For an array of N elements indexed by $i \in \{0, 1, ..., N-1\}$, the effective π approximation at position i is given

by:

$$\pi\blacksquare(i) = D[i \bmod L]$$

This substitution is applied per-element across any formula that would ordinarily use π. For example, in sinusoidal array generation, the standard expression sin(2π · i / N) becomes sin(2 · π■(i) · i / N), with no floating-point values required for the π term itself.

2.2 Connection to 22/7

The arithmetic mean of the sequence D is:

$$(2 + 2 + 2 + 2 + 8 + 4 + 2) / 7 = 22 / 7 \approx 3.142857$$

This is precisely the classical Archimedean approximation of π, accurate to within 0.04% of the true value (3.14159...). The sequence D therefore encodes 22/7 implicitly — not as a fraction requiring division, but as a repeating integer pattern whose mean reproduces the approximation across any complete cycle.

This equivalence was arrived at intuitively through empirical search, without prior knowledge of the 22/7 relationship — a result that lends the method independent validity as a rediscovery of a classical result via a novel encoding.

2.3 Structural Properties

Several structural properties of the method are worth noting. First, the approximation quality is cycle-coherent: any contiguous sub-array whose length is a multiple of L will have a mean π approximation equal to 22/7, regardless of starting offset. Second, the method introduces controlled per-sample deviation — positions assigned the digit 8 experience the largest local error, while positions assigned 2, 3, or 4 remain closer to the true value. This per-sample error is deterministic and predictable, which distinguishes it from noise-based or stochastic approximation schemes.

Third, the method is trivially extensible: longer sequences with more digits summing to a larger numerator over a larger denominator can encode better rational approximations of π, such as 355/113 (accurate to seven decimal places), at the cost of a longer cycle period.

3. Application Context

The primary motivation for this method is float-free computation. In environments constrained to integer arithmetic — including many embedded DSP pipelines, retro computing platforms, and low-level bare-metal systems — the method provides a lightweight mechanism for introducing π-like behavior into array operations without requiring a floating-point unit, division hardware, or precomputed lookup tables.

A secondary application is in generative and procedural signal contexts, where the time-varying nature of π■(i) produces deterministic but non-stationary waveforms with predictable spectral properties. The deliberate introduction of per-sample frequency variation creates waveforms that differ structurally from both pure sinusoids and noise, occupying a novel region of the signal design space.

4. Limitations

The method has two principal limitations. The first is per-sample accuracy: individual array elements computed with digit 8 as their π approximation experience significant local error, as 8 is not a plausible approximation of π at the per-sample level. Applications requiring high per-sample fidelity should prefer fixed-point methods.

The second limitation is phase coherence in sinusoidal applications. Because each sample is computed independently using its local π■(i), no phase accumulation occurs across samples. This introduces discontinuities at digit transitions that would manifest as spectral artifacts in audio or signal processing contexts. Phase-accumulation variants of the method, where the cyclic digit modulates a running phase rather than the per-sample argument, would address this at the cost of added implementation complexity.

5. Conclusion

This paper has described a novel method for approximating π as a time-varying cyclic integer sequence, demonstrated its equivalence to the classical 22/7 rational approximation, and outlined its properties and limitations. The method is original in its framing — treating π not as a constant but as a signal — and is practically motivated by the requirements of float-free integer computation.

Future work may explore optimal digit sequences encoding higher-precision rational approximations, phase-accumulating variants for smooth sinusoidal generation, and the spectral characterization of waveforms produced by this method.